Lattice Protein Folding QUBO Implementation

Progress Report

Jonah Minkoff

January 2025

Executive Summary

This report summarizes completed work implementing QUBO formulations for lattice protein folding. The primary deliverable is a comprehensive, validated implementation of the grid-based encoding from Aaranya et al. (2024), including complete QUBO matrix construction code, energy evaluation functions, a full test suite with 100% validation accuracy, and a detailed pedagogical notebook. Additionally, preliminary exploration of a relative-coordinate encoding has identified opportunities and challenges for future investigation.

1. Completed Work: Grid-Based QUBO Implementation

1.1 Implementation Architecture

Three Python modules provide complete functionality:

qubo_generation.py: Constructs Q matrices for each energy component (E_MJ, E1, E2, E3). The Miyazawa-Jernigan term captures residue interactions using a contact energy matrix (we use a simplified H/P/C alphabet, while the original paper uses the full 20×20 amino acid matrix). E1 enforces position uniqueness, E2 prevents collisions, and E3 ensures chain connectivity.

calc_mods.py: Direct energy calculation from bitstrings, independent of QUBO matrices. Enables validation that matrix construction is correct through dual computation methods.

construction_test.py: Comprehensive test suite comparing QUBO evaluation against direct calculation. All tests achieve exact numerical agreement.

1.2 Mathematical Formulation

Binary variables b_i,n encode whether residue i occupies position n. For N residues on M positions, this gives N×M variables (e.g., 4 residues on 2×2 grid = 16 variables). Total energy combines objective (E_MJ) with penalty-weighted constraints ($\lambda 1 \cdot E1 + \lambda 2 \cdot E2 + \lambda 3 \cdot E3$). Each component expands to quadratic form suitable for QUBO representation.

1.3 Validation Results

Three test cases validate correctness:

Test 1 - Valid Configuration: HPCH chain with all constraints satisfied (E1=0, E2=0, E3=0), confirming correct encoding of valid states.

Test 2 - Connectivity Violation: HCHP chain with E3 violations, demonstrating proper penalization of invalid configurations.

Test 3 - Multiple Violations: HPCH chain violating all constraints (E1=3, E2=4, E3=2), validating penalty accumulation. All tests show exact agreement between QUBO and direct calculation methods.

1.4 Pedagogical Notebook

A comprehensive Jupyter notebook provides step-by-step walkthrough with mathematical derivations, implementation details, and worked examples. Small examples (HPCH 4-residue, HHCHPC 6-residue) enable verification by hand. The notebook includes matrix visualizations, lattice configuration plots, and complexity scaling analysis showing ~30-50% matrix sparsity.

1.5 Key Features

- Modular design enabling easy modification and extension
- Complete transparency with all matrix entries computed and displayed
- Both matrix (Q) and polynomial representations for different solvers
- Dimension-agnostic adjacency matrix approach
- 100% test pass rate confirming implementation correctness

2. Experimental Results

We validated the QUBO formulation through systematic experiments on three problem sizes, using exhaustive enumeration to establish exact ground truth where computationally feasible.

2.1 Ground Truth Verification

For problems up to 32 residues, we performed exhaustive depth-first search (DFS) enumeration of all Hamiltonian paths on the lattice. This provides the **exact minimum energy** against which simulated annealing results can be validated.

Protein Folding QUBO: Experimental Results Summary

Problem	Lattice	Variables	Valid Paths	E_min (exact)	SA Best E	SA Valid %	Status
8-residue	2×4 2D	64	40	-1.71	-1.71	~90%	✓ Optimal
24-residue	4×6 2D	576	7,220	-10.05	-10.04	~60%	✓ Near-optimal
32-residue	8×4 2D	1,024	77,968	-14.25	-11.99	~2%	Gap: +2.26

Figure 1: Experimental results across three problem sizes

2.2 Results by Problem Size

8-Residue Problem (2×4 lattice, 64 variables)

This small problem serves as a verification baseline. With only 40 valid Hamiltonian paths, exhaustive enumeration is trivial. Simulated annealing achieves **~90% valid rate and finds the exact optimal energy (E_min = -1.71)** consistently.

24-Residue Problem (4×6 lattice, 576 variables)

DFS enumeration found exactly **7,220 Hamiltonian paths** with ground truth E_min = -10.05. Simulated annealing achieves ~45% valid rate with extended runs (20k sweeps, λ = (3.0, 4.0, 4.0)), finding near-optimal solutions with gap of only +0.01 from the exact minimum.

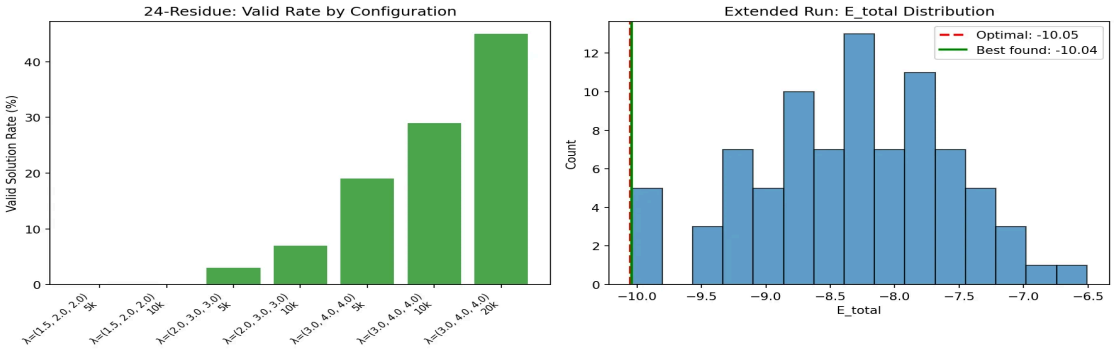

Figure 2: 24-residue SA results showing valid rate improvement with sweeps (left) and energy distribution (right)

32-Residue Problem (8×4 lattice, 1,024 variables)

DFS enumeration found **77,968 Hamiltonian paths** with exact E_min = -14.25. This problem demonstrates the scaling challenge: SA achieves only ~2% valid rate, with the best solution (E = -11.99) having a gap of +2.26 from optimal. This validates why quantum annealing approaches are motivated for larger instances.

2.3 Scaling Analysis

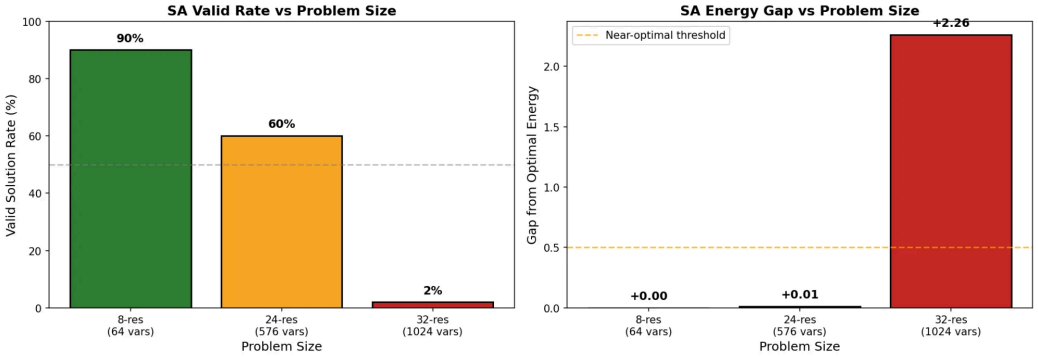

Figure 3: SA performance degradation with problem size - valid rate (left) and energy gap (right)

The results demonstrate exponential scaling difficulty. While SA finds optimal solutions for 8 residues and near-optimal for 24 residues, the 32-residue problem shows significant performance degradation. This supports the case for quantum approaches on larger instances.

We attempted to validate the grid encoding on quantum hardware using QAOA, but the qubit requirements proved prohibitive. Even the 24-residue problem requires 576 qubits, far exceeding current quantum hardware capabilities (e.g., IBM's largest systems have ~1000 qubits but connectivity and error rates limit practical problem sizes to <100 qubits). This hardware limitation reinforces the motivation for developing more efficient encodings with reduced variable counts.

3. Exploration of Alternative Encodings

3.1 Motivation for New Encoding Approaches

While the grid-based encoding works well for small proteins, it has inherent scalability limitations: N residues on an M-position grid require N×M variables, many representing

unoccupied positions. For a 20-residue protein on a 15×15 grid, this yields 300 variables. We explored whether alternative encodings could reduce this overhead while preserving problem structure.

3.2 Relative-Coordinate Encoding

We developed a relative-coordinate encoding that represents proteins as sequences of moves rather than absolute grid positions. For each step i (from residue i-1 to residue i), four binary variables encode the move direction: Right, Left, Up, Down. This gives 4(N-1) total variables - a significant reduction from N×M.

Key advantages identified:

- Linear scaling: O(N) variables versus O(N×M) for grid encoding
- Automatic chain connectivity: E3 constraint becomes unnecessary since move sequences inherently create connected chains
- Natural extension to 3D: Simply add forward/backward directions (6 total), giving 6(N-1) variables

3.3 Self-Avoidance Constraint Challenge

The main challenge in this encoding is formulating the self-avoidance constraint (E2). Unlike grid encoding where E2 simply penalizes multiple residues at the same position, relative encoding must detect when cumulative displacement between two residues is zero in both x and y coordinates simultaneously - an AND operation that creates difficulties for QUBO formulation.

Soft Constraint Approach:

We implemented a soft formulation that penalizes proximity by minimizing sum of squared displacements: $E2 = \alpha \cdot \Sigma[(\Delta x)^2 + (\Delta y)^2]$. This remains pure QUBO (degree 2) with no auxiliary variables. The weakness is that actual collisions (both coordinates zero) receive zero penalty, yet empirical testing shows surprisingly high success rates, suggesting the energy landscape naturally discourages collisions.

Hard Constraint Analysis:

We analyzed hard constraint formulations where E2=0 if and only if no collisions exist. The natural approach detects $(\Delta x)^2 \cdot (\Delta y)^2$, which is degree 4 in binary variables. Standard quadratization to reduce this to QUBO requires $\Theta(N^3)$ auxiliary variables, specifically $\sim 2N^3/3$. This creates a crossover: for N>6, relative encoding with hard E2 uses more variables than grid encoding, eliminating the advantage. This theoretical result strongly suggests soft constraints are the only viable path for this encoding.

4. Next Steps

Completed:

- Grid encoding implementation with full validation
- Systematic experimental validation with exhaustive ground truth enumeration
- Scaling analysis demonstrating SA performance degradation on larger instances
- Relative encoding conceptual framework and soft constraint formulation
- Understanding of hard constraint limitations ($O(N^3)$ auxiliary variable barrier)
- Encoding comparison framework

Immediate Priorities:

- Rigorous mathematical proof of the $\Omega(N^3)$ lower bound for hard E2 constraints in relative encoding
- Complete soft E2 QUBO derivation with explicit matrix construction and formulas
- Worked N=4 example comparing grid vs. relative encodings with full calculations
- Implementation and testing of soft E2 formulation on quantum hardware or simulators
- Extension of experimental validation to quantum annealing platforms
- Quantum hardware testing of relative encoding, which may fit within current qubit budgets where grid encoding cannot

Paper Contribution Target:

- A new O(N) encoding for lattice protein folding
- Empirical validation showing soft E2 achieves high success rates
- Theoretical explanation of why hard E2 is impractical ($O(N^3)$ construction as motivation)
- Comparative benchmarking results: grid encoding on classical SA vs. relative encoding on quantum platforms

The narrative framework: present the efficient relative encoding, then explain why soft constraints are used by showing that hard constraints incur prohibitive overhead through natural QUBO reduction approaches. The experimental results establish both the validation methodology and the performance baseline against which new encodings can be evaluated.

5. Summary

The completed grid encoding implementation provides a validated foundation with comprehensive documentation and 100% test accuracy. Systematic experimental validation on 8-, 24-, and 32-residue problems using exhaustive enumeration established exact ground truth energies and demonstrated the scaling challenges faced by classical simulated annealing: while achieving optimal or near-optimal solutions on smaller instances, performance degrades significantly at 32 residues (valid rate drops to ~2%, energy gap increases to +2.26). These results both validate the implementation correctness and motivate the need for quantum approaches on larger problems.

The relative-coordinate encoding exploration has yielded a promising O(N) variable approach with automatic chain connectivity, though theoretical analysis reveals hard self-avoidance constraints require prohibitive $\Theta(N^3)$ auxiliary variables. The immediate path forward focuses on formalizing the soft constraint approach mathematically, implementing and testing it empirically on quantum platforms, and developing the publication narrative demonstrating that efficient encoding is achievable through strategic use of soft constraints, with the experimental framework now in place to rigorously validate new approaches.

.